



Consorzio per la Formazione e la Ricerca
in Ingegneria dell'Informazione

Politecnico di Milano

XIV Master in Information Technology
Master thesis abstract

Intrusion Detection Systems and Log Correlation

Author: *Carmelo Silvestro*
Tutor: *Davide Cerri*
Unit: *e-Service Technologies*
Sponsor: *Key Consultants*

July 5, 2002
Version: *1.0*
State: *deliverable*

Abstract

A recent study by Computer Security Institute (CSI) has pointed out that about 70 per cent of organizations, monitored in the last year, has reported a security incident. The nature of e-business requires that organizations open their networks to the Internet, but security issues must be taken into account. Computer security is an expanding field of information technology, and in the very next future large investments will be made.

Attacks to information take advantage of system and protocol vulnerabilities. System vulnerabilities fall in three classes: *software bugs* (e. g. buffer overflow); *system configurations* (default configurations are often a backdoor to gain access in computers, trust relationship with compromised hosts...); *password cracking* (e. g. brute force attacks). Protocol vulnerabilities are caused by weaknesses in design of TCP/IP stack or by bad implementations.

In order to protect information systems from intruders, intrusion detection systems are becoming fundamental because human surveillance alone is ineffective. IDSs (Intrusion Detection Systems) are automatic hardware devices or software agents that monitor events (network traffic, data audit, application logs) in computer systems or networks and their goal is to alert the security manager. The most common classification of IDSs groups them by the kind of information source (network packets, logs,...).

If the IDS analyzes network packets from cables we call it *network-based* (*NIDS* – Network Intrusion Detection System): most commercial tools are network-based IDSs because they can monitor a large network (network traffic directed to multiple hosts). *Host-based* IDSs (*HIDS* - Host Intrusion Detection Systems) analyze information data from a single host: they are more accurate, and they can determine if the exploit compromised the target operating system.

The two approach of events' analysis are:

- *misuse detection* (the engine looks for events that match a defined attack pattern);
- *anomaly detection* (the detector has the knowledge of “normal” activity and an attack can be detected if it is different from legitimate activity).

IDSs will have a distributed capability to administer some sensors and agents for deployment in effective large networks. A recent trend in IDS research is the field of log correlation, because the IDS must integrate different kinds of logs from heterogeneous sources. *Correlation*, in fact, is the ability to determine a pattern of an attack scenario among atomic events. Now, to cover large networks, sensors can be placed throughout the network, but a new problem arises in collecting data from different sensors. Correlation allow us to deduce knowledge from single security events. Some advantages of correlation are:

- false alarm reduction;
- detection rate growth.

Commercial tools provide simple aggregation or data fusion (examples are *Tivoli* and *netForensics*): events from multiple sources are collected by some fields (source IP address, destination IP address, attack class) and displayed at the management console. The main architectural approach is central management: a central unit or engine receives the alert information from many sources (firewall, NIDS, HIDS, web server, etc.) and centrally processes them.

In this Master thesis we examined the context of real time correlation and we implemented a prototype of correlation engine in a distributed environment but with a centralized engine: we chose an expert system (CLIPS) because rule-based programming allows knowledge to be represented at higher levels of abstraction. The sources of our system are several agents which run over the sensors: they send only the relevant events at higher level in a predefined format. We wrote some aggregation rules like those of Tivoli commercial tool: for example a rule groups the events with the same source (intruder), the same target (victim) and same class of alert (attack signature) and displays alert in a compact manner. We also defined new correlation rules.

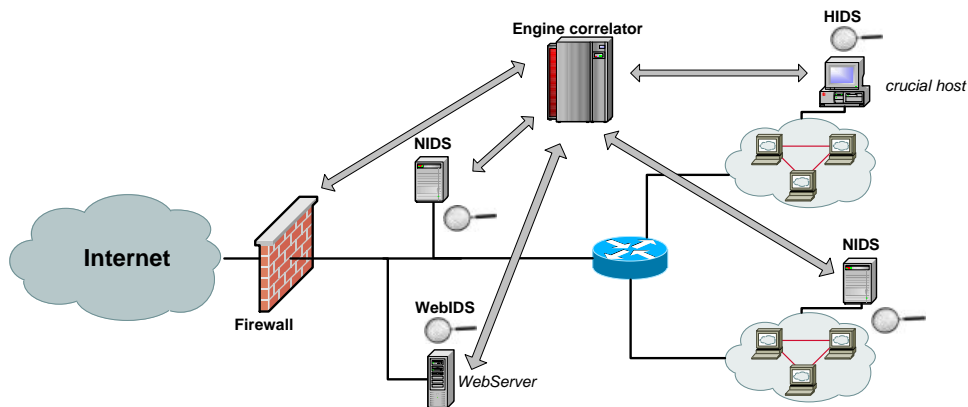


Figure 1: Centralized architecture for correlation